MASSACHUSETTS INSTITUTE OF TECHNOLOGY
Electronic Systems Laboratory
Cambridge 39, Massachusetts

M E M O R A N D U M

M-5001-21

TO: J. T. Dennis

FROM: J. E. Ward

SUBJECT: FLEXO NOISE CHECK 4-60

DATE: 22 April 1960

---

1. PROGRAM OPERATION

Turn Type-in switch on and prime flexowriter. Read in UT-3 (5600 version) first, followed by FLEXO NOISE CHECK. Program will print sequence of carriage returns and tabs and halt. If any flexo noise occurs on any print operation, the contents of the delay counter for that operation will be typed in octal. If no noise occurs, nothing will be typed. The sequence can be repeated, if desired, by pushing restart.

2. BACKGROUND

In the past, a number of people have experienced difficulty with "flexo noise," which manifests itself as a spurious input to the TX-0 Live Register from the on-line Flexowriter when printing out. About a year ago I investigated this (of necessity, since my program was being buggered), and found that only two flexo print operations, tab and carriage return, caused Live Register inputs. This was determined by Test Mode print mode operations from the console. It was found that printing tab or carriage return caused the flexo code for these same operations to be placed in the Live Register.

The next question was to determine the time delay in this feedback from the Flexo Circuits. Since the outgoing code for carriage return is 101001, and the signal returned to the IR is 644000, it is obvious that the flexo type-in circuits are somehow energized during the print cycle. Since these are relay circuits, a long time delay (compared to TX-0 instruction times) is to be expected. The FLEXO NOISE CHECK program was written to measure the time delay following tabs and carriage returns before the IR input occurs.

It was originally intended that the timing information thus obtained would be used to design a standard delay loop which could be used as a macro instruction to clear the IR after tabs and carriage returns. Since an electrical delay has now been provided in the TX-0 which prevents the IR inputs, the program is useful to check that cancellation indeed occurs.

3. PROGRAM DESCRIPTION

The program, shown in flow diagram form in Fig. 1, prints a sequence of tabs and carriage returns, clears the LR, and then "listens" to the LR after each print until the spurious input occurs, or until a time limit is reached. If a spurious input occurs, the program uses the UT-3 printout routine (in 5600 location) to print the octal content of the loop counter. The listen loop time is 90 μsec, and this is to be multiplied by the decimal loop count to obtain the time delay.

It was originally anticipated that the delay might be a function of the carriage position, so a format was chosen to test this possibility. Thus the program prints an initial carriage return, and then prints a variable number of tabs from 7_{10} down to 1, with each group followed by a carriage return. A tab counter in the program is initially set to -7, and this is reset to successively smaller values after each tab group. The program halts when the tab counter reset value is positive, and will recycle the whole operation if restarted.

The original version of the program was looking for an input which always occurred and no other means was provided to terminate the listen loop. Since LR inputs are now the exception rather than the rule, the program has recently been modified to include a "time" limit on the loop count, after which it will skip the print of the loop count and proceed to the next tab or carriage return in the sequence. No typeout means no flexo noise! The time limit chosen, 4000_8 loops, is about 4 times the maximum delay measured (see next section). If a longer time is required, it may be stored as -N in register "lim" (register 163 in binary tape).

An English copy of the program is appended. Possible improvement as a noise check could be made by adding a section to print all possible flexo codes from 0 to 77, with a "listen" cycle after each one.

4. DELAY MEASUREMENTS

Two typical printouts obtained on May 18, 1959 are shown below. The first number is the loop count for the first carriage return, the second number for the first tab, etc. It will be noted that for this particular flexo, the times for tabs are somewhat longer than for carriage returns, and that both times are reasonably uniform (about ± 6 o/o) and independent of carriage position.

635	671	733	742	762	753	756	737
644	711	703	677	735	745	763	
651	705	742	751	753	731		
617	726	702	670	716			
620	744	707	705				
575	733	703					
605	702						
642							

652	727	710	713	725	733	741	735
640	741	725	713	714	711	710	
577	701	701	714	723	727		
636	735	762	747	721			
617	722	742	750				
644	733	740					
623	734						
624							

The actual time delay can be computed by converting the above octal numbers to decimal and multiplying by 78 μsec, the loop time of the program in use for these measurements. The maximum delay shown is 763_8. To add a slight factor of safety, assume a delay of 1000_8, which corresponds to 512_{10}. Maximum delay for the flexowriter tested is thus about

$$512 \times 78 \times 10^{-6} = 40 \times 10^{-3} \text{ sec}$$

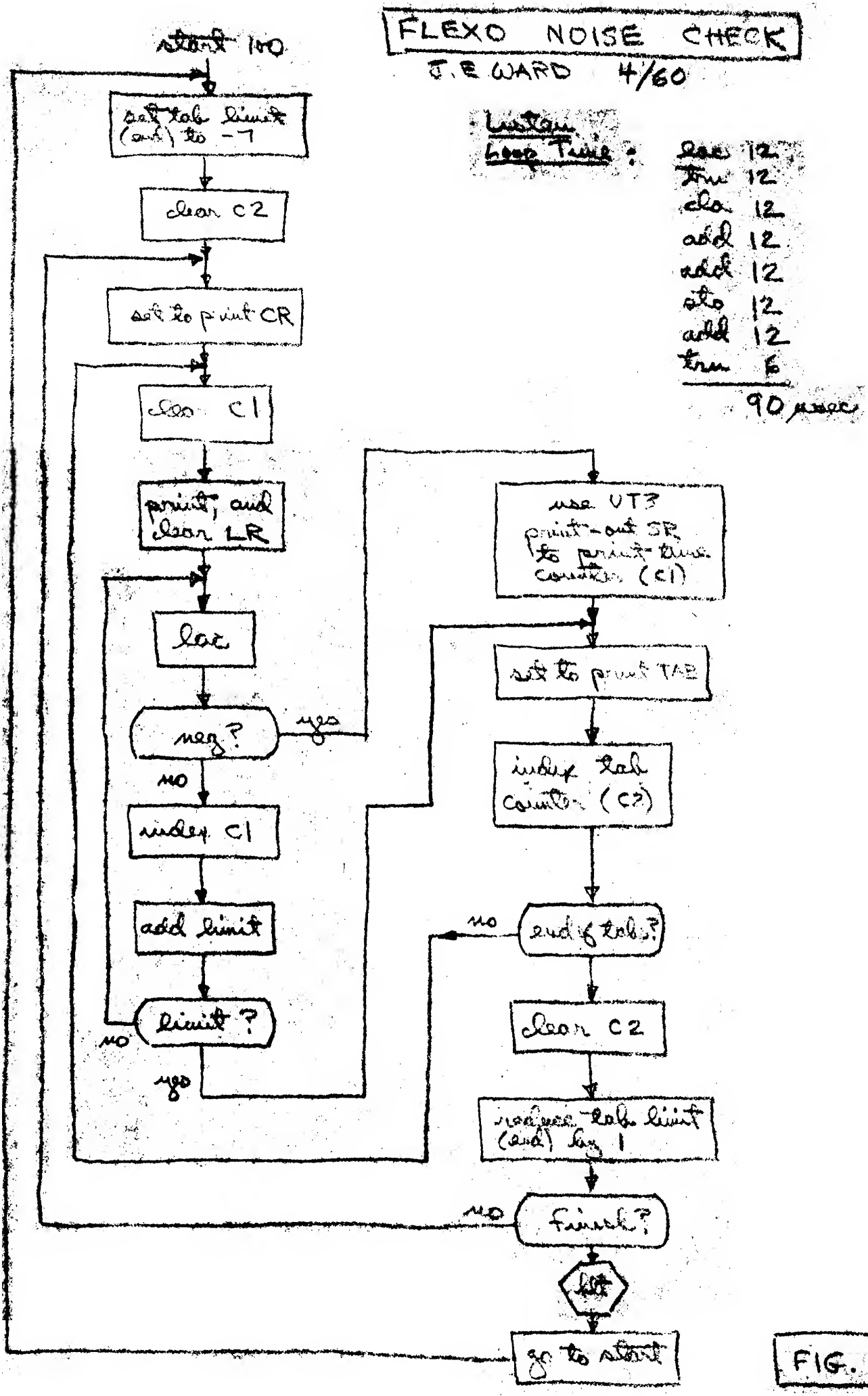
which means the noise enters the LR up to 3600_{10} instructions after the offending print!

John E. Ward
Assistant Director

JEW/pem

FLEXO NOISE CHECK

T. E WARD 4/60



Instruction Loop Time:

lac	12
tru	12
cla	12
add	12
add	12
sto	12
add	12
trn	6
	90 μsec

FIG. 1

```
flexo noise check 4-60

define
        load A
                cla
                add A
        terminate

define
        clear A
                cla
                sto A
        terminate

define
        index A
                cla
                add A
                add (1
                sto A
        terminate

define
        print A
                cla         | use 5600 ut-3 for printout
                add A
                sto 5724
                cla
                add .+z
                trn 6256
        z,      trn .+z+1
        terminate
```

```
Constants

        164         to          170

Defined symbols

z=6
a=105
c=110
b=112
e=114
d=125
f=134
c1=156
c2=157
cr=160
tab=161
end=162
lim=163
```

```
                    sto 5724
                    cla
                    add .+z
                    trn 6256
          z,        trn .+z+1
          terminate


tra=500000

 100      lcad (-7              | set tab limit
          sto end
          clear c2
 a,       load (add cr          | set for cr
          sto b
 c,       clear c1
 b,       add                   | cr or tab
          prt+200
 e,       lac
          trn d
          index c1              | count until lr is neg or time limit
          add lim
          trn e
          tra f
 d,       print c1              | print loop count
 f,       load (add tab         | set for tab
          sto b
          index c2              | check tab count
          add end
          trn c
          clear c2
          index end             | reduce tab limit
          trn a                 | stop if tab limit is pos
          hlt
          tra 100               | recycle


 c1,      0                     | time count
 c2,      0                     | tab count
 cr,      101001
 tab,     100101
 end,     0
 lim,     -4000                 | limit on time count

constants

start 100

x
```